



# APP DEV: SongSearch

## Contents

- 2. Concept
- 3. Research Contexts
- 5. User Personas
- 6. Inspiration
- 7. Initial Designs
- 8. Wireframes
- 9. Key Frames
- 10. Code
- 11. Final Designs
- 13. Limitations / Challenges
- 14. References

# Concept

**SongSearch is a mobile app designed to bring together music enthusiasts from diverse backgrounds, providing a platform for users to connect, engage in lively discussions, and discover new artists, songs, and events to go to. The app offers a range of features, including chat rooms, personalized profiles, song recommendations, music events, and social sharing capabilities. By fostering a vibrant community of passionate music lovers, SongSearch aims to enhance the music discovery experience, encourage meaningful interactions, and expand users' musical horizons. Through user engagement metrics and community growth tracking, the social impact of SongSearch can be measured, assessing its role in promoting music appreciation, fostering connections, and empowering users to explore and share their favourite musicians and songs.**

**The community structure within SongSearch is decentralised and user-centric. While the app offers features like chat rooms and personalised profiles, it empowers users to generate content and engage with others based on their shared musical interests. There is no formal hierarchy within the community, allowing users to freely connect, discover, and interact with fellow users.**

# Research Contexts

## Problems with existing music streaming services

Music streaming algorithms aren't working to allow users to find new music and artists. A study by Werner in 2016 focused on three popular music genres: pop, rock, and hip-hop. Using a sample of Spotify users, the authors collected data on the gender of the artists recommended by the platform. They compared the actual gender distribution of artists in each genre to the gender distribution of recommended artists. The findings of the study revealed a significant gender bias in Spotify's music recommendations, with male artists found to be overrepresented in the recommendations (Werner, A., 2020). Another study conducted in 2019 studied the relationships between users, automated curation and recommendation systems on music streaming platforms (Freeman, S., Gibbs, M., & Nansen, B. (2022)). The positives include users being able to find new music, however, users questioned who held expertise in music taste: themselves or the algorithms. Overall the users preferred conversational chats between the user and the algorithm. Additionally, users have limited control over algorithmic filtering (Freeman, S., Gibbs, M., & Nansen, B. (2022)). Despite the numerous benefits associated with the emergence of artificial intelligence (AI), it has been observed that users still have a preference for conversational-like recommendations. This suggests that user-centric recommendations continue to be essential in meeting user preferences and needs.

## Limitations with existing music apps

Spotify Radio, despite claiming to offer personalised and never-ending music, actually delivers songs in limited loop patterns. Researchers aimed to investigate the characteristics of these loop patterns and determine whether they are finite or infinite. To conduct the experiment, 160 bot listeners, all Spotify Free users, were programmed to listen to different Swedish music from the 1970s. These bots documented the subsequent tracks played in the radio loop and interacted with the Spotify Web client by liking, disliking, and skipping songs (Snickars, P. (2017)). Firstly, it was found that loop patterns could be measured on Spotify Radio, indicating that the music delivery platform operates within specific repetitive patterns. The loops display a repeated pattern with slight variations based on the artist or bot characteristics. The study confirmed that Spotify Radio's music loops are not endless but rather consist of a finite number of tracks. The research also showed the algorithms tend to offer slight variations of what users already consume, rather than expanding their musical horizons. The experiment further revealed that personalisation through user recommendations, such as liking, disliking, or skipping songs, did not significantly impact the results of Spotify Radio. Despite claims that personalizing stations would improve the music selection, the study found little evidence to support this. In contrast to these features that rely on AI algorithms for music recommendations, my app highlights a unique feature that empowers users to suggest new songs. By emphasising user-generated content and community engagement, my app gives a platform where users have an active role in shaping the music discovery experience.

## Requirements for a new music discovery app

A study conducted in Finland in 2013, compared five music player prototypes designed to discover new music. The research aimed to create innovative and user-friendly interfaces, evaluate the prototypes with real users, and find design implications for music discovery applications. The study involved 40 Finnish participants who tested five prototypes; an avatar-based user interface for discovering new music, singing animated mood pictures for music recommendations, a cube-based user interface for exploring music collections, a potentiometer-based graphical user interface for interacting with a music recommendation service, and an album cover space preview for recommendation playlists. The results indicated that visually-interesting interfaces and new interaction paradigms inspired users to explore music collections and actively seek new music. (Lehtiniemi, A., & Holm, J. (2013)). The prototypes were considered easy to use and provided playful ways to find new music. The album cover space prototype was the overall winner, with positive feedback for its "Get similar" function and wide variety of features and genres. The study formulated criteria for designing music discovery services, such as minimizing text and menus, providing search functionality, supporting related information, and integrating social networking. Additionally, investigating how visual music player interfaces can complement traditional music player applications is another interesting area for research. Overall, the study supports the need for an app that focuses on music discovery by emphasising the importance of user-friendly interfaces, novel approaches, complementary features, and incorporating user feedback. This is a large part of my app design proving there is a need for an app like mine.

# Research Contexts

## Importance of UI/UX Design

Statistical data reveals that the initial impression of an app plays a significant role in determining user satisfaction, with 94% of the satisfaction index being influenced by the first exposed design elements, such as the app icon and intro screen. While design alone does not guarantee app success, it undeniably plays a crucial role in impressing users. When it comes to app design, there are two main focuses: UX (user experience) and UI (user interface). UX aims to enhance customer satisfaction by ensuring optimal usability and user comfort, emphasizing familiarity (Cha, M. H., & Woo-Kyung, L. (2021)). On the other hand, UI focuses on aesthetics, including elements like colour schemes and visually appealing animations. One notable design trend is the widespread adoption of dark mode, which gained popularity with the introduction of iOS 13 in 2019. Dark mode is essentially “a dark-grey colour scheme” (Pogue, 2020, p. 152) that aims to alter the background colour of an application to black. The rise in popularity of dark mode can be attributed to its associated health benefits and positive impact on well-being, as it helps reduce eye strain and saves battery power (Hakobyan, L., & Saha, R. (2021)). Dark mode has emerged as an essential aspect to be taken into account in digital marketing communication. The integration of dark mode has become more prevalent in the development of social media applications (Pedersen et al., 2020, 178). This feature has continued to be favoured by users, prompting further research into its visibility and implementation costs.

Additionally, music apps such as Spotify have an effective UX design, the navigation bar, home page, and "explore" page use colourful fonts and designs which are visually appealing against the dark background and also have transitions that make the user feel like they're in control (Besseney, A.). The balance of these buttons against a simple dark interface prevents user frustration and ultimately makes the app simple to use without prior expertise from the user (Jackson, M. S. (2023). Spotify: The Addiction).

## Other Notable features include:

- **Vibrant Colour Schemes:** A wide range of colour options and the bold contrast of contrasting colours offer a visually captivating experience to users.
- **Distinctive Font Styles:** In iOS 11, larger and bolder title fonts are employed to emphasize central content, distinguishing it from other elements (Cha, M. H., & Woo-Kyung, L. (2021)).
- **Ample Rounded Corners:** The use of larger fillets and increased margins creates a more spacious and transparent visual experience.
- **Engaging Illustrations:** UI illustrations have become more diverse in terms of style and content. The popularity of 3D-style illustrations has risen, while the narrative illustration approach has gained traction for conveying content effectively.

# User Personas



Jack - 19

**About:** Jack is a university student who finds himself disconnected from the typical student nightlife and struggling to fit in with the crowd. Having moved to a new city for his studies, he faces the challenge of exploring and discovering places that align with his interests. With a strong passion for live music and a love for attending concerts, Jack seeks opportunities to connect with like-minded individuals who share his enthusiasm for the vibrant music scene.

**Solution:** Through SongSearch, he can discover upcoming concerts and performances in his new city, ensuring that he never misses out on the thrilling experiences of live music.

In addition, SongSearch empowers Jack to create group chats, allowing him to connect with fellow concert-goers and music enthusiasts. These group chats serve as a platform for fostering connections, sharing concert experiences, and even planning outings to upcoming events.



Kelly - 24

**About:** Kelly is a young woman in her 20s with a job as a marketing assistant. She has a strong interest in music and is always on the lookout for new artists and songs to add to her playlist. However, her social anxiety often makes it difficult for her to engage in face-to-face interactions and establish meaningful connections. Kelly finds solace in online communication, where she feels more at ease expressing herself and connecting with others.

**Solution:** By utilising SongSearch's chatrooms, Kelly can connect with fellow music enthusiasts, share her favorite artists, and discover new recommendations. The app's private messaging feature allows her to initiate conversations and gradually become more comfortable in interacting with others.

Furthermore, SongSearch's personalised recommendations play a crucial role; by considering her musical preferences and social anxiety concerns, the app suggests smaller, more intimate music events in her area.



Davide - 31

**About** - Davide is a passionate music event organiser in his early 30s. He loves curating unique and memorable music experiences for music lovers in his city. Davide faces the challenge of effectively promoting his music events and reaching a wider audience. He wants to connect with music enthusiasts who would appreciate the events he organises. Additionally, he struggles to keep track of local artists, their availability, and their potential to fit into his event lineup.

**Solution** - The app offers a platform for David to promote his music events to a targeted audience of music enthusiasts. He can create event listings, share event details, and engage with potential attendees through chatrooms and private messaging.

Additionally Davide could discover local emerging artists that could he could further work with. He can access their profiles, listen to their music, and evaluate their suitability for his event lineup.

David can connect with other event organisers and industry professionals within the SongSearch community. This enables him to collaborate, share insights, and expand his network for future event opportunities.

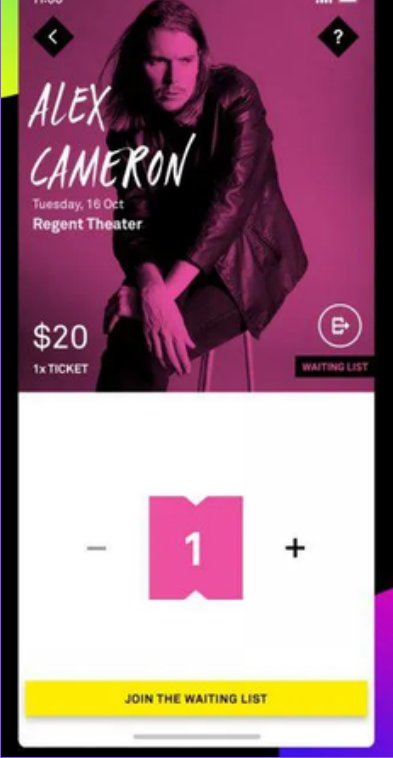
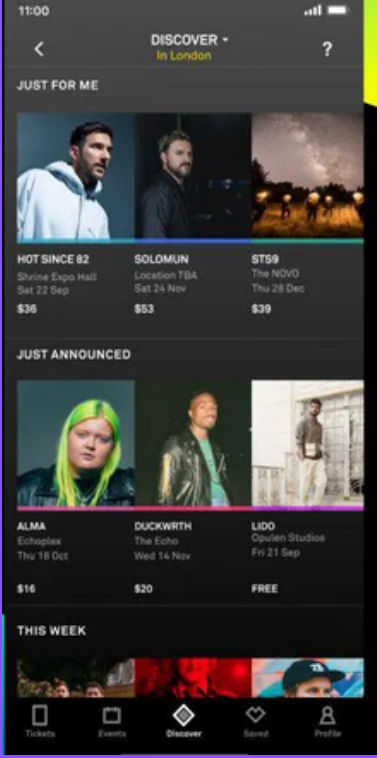
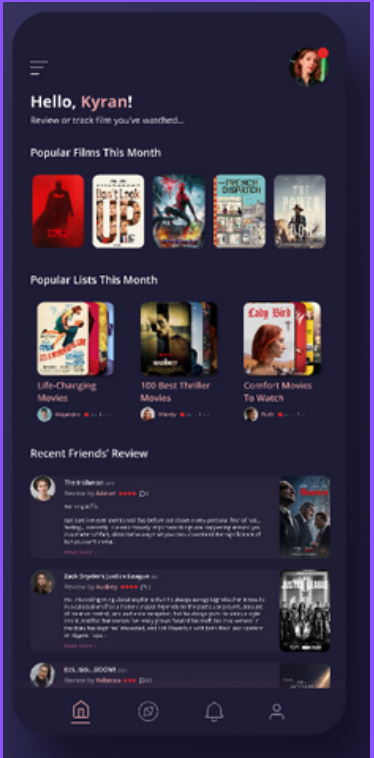
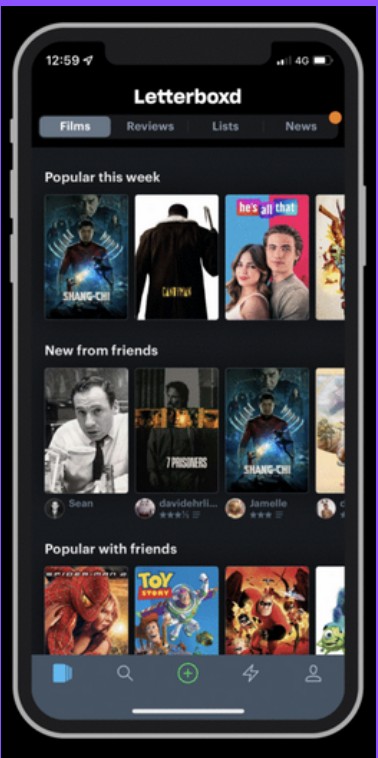
# Inspiration

To design my app, I embarked on a thorough exploration of existing social media platforms and their features that could seamlessly integrate into my vision. During my research, I discovered several notable apps that served as valuable sources of inspiration. One such app was Letterboxd, a popular film discussion platform with large socialising capabilities and extensive user personalisation options. Drawing inspiration from Letterboxd, I sought to create a similar sense of community and interaction within my app.

Additionally, Pinterest's grid theme gained my attention due to its visually appealing layout and efficient organisation. I recognised the potential of incorporating a collection feature into my app, allowing users to curate their own personalized content. By drawing inspiration from Pinterest's grid design, I aimed to create a visually engaging and intuitive user experience, enhancing the app's aesthetic appeal and ease of use.

Another influential app was Dice, which facilitates ticket purchases and event sharing among friends. Recognising the importance of seamless event discovery and social engagement in my app, I sought to incorporate similar features from Dice. By integrating event ticketing capabilities and social sharing functionalities, I aimed to create a comprehensive platform that allows users to effortlessly explore, attend, and share exciting music events with their connections.

In terms of visual design, I found inspiration in Spotify's dark theme. The captivating aesthetic and sleek interface of Spotify's dark theme prompted me to adopt a similar approach in my app. By utilising a dark theme, I aimed to create a visually striking environment where albums, events, and other content could stand out, while also incorporating rounded effects on images to add a modern and elegant effect.



**Letterboxd**

- Reviews and chat rooms
- Layout
- Users Personification - users can curate lists and collections

**Dice**

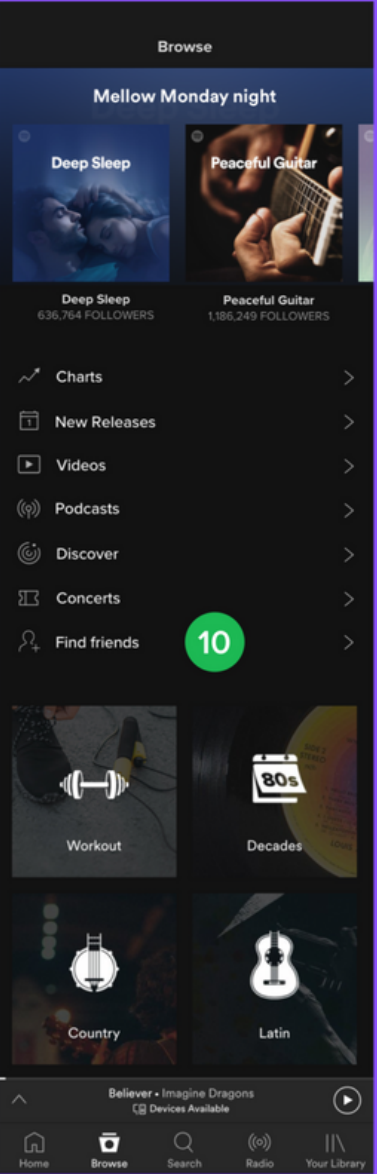
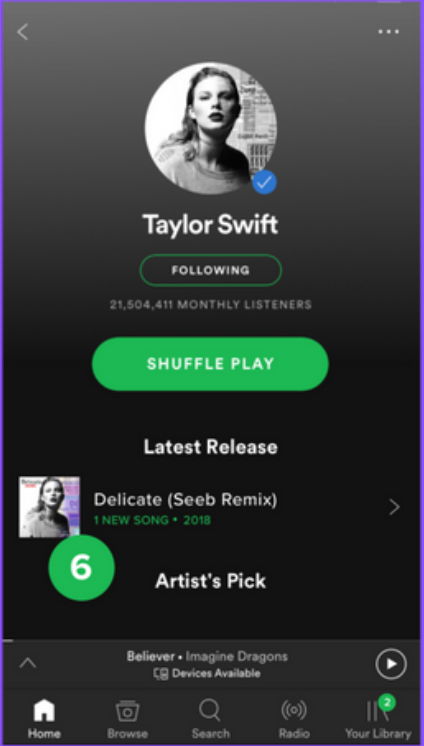
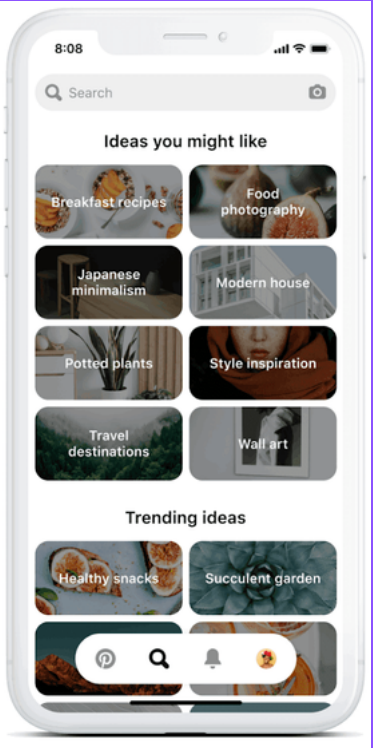
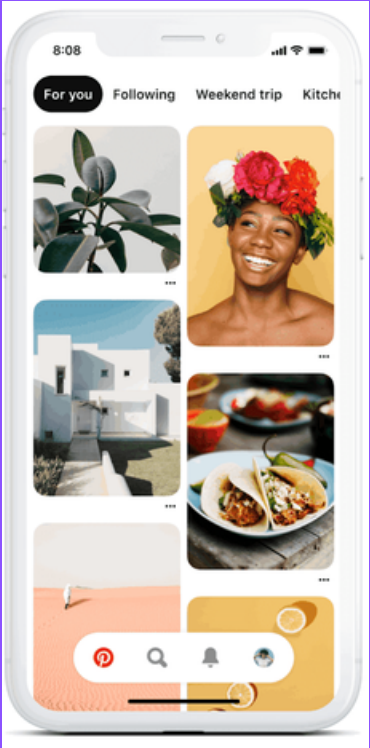
- Colour pallet
- Events Features
- Sharing Features

**Pinterest**

- Layout of collections and photos
- Curation of pictures
- Save Feature

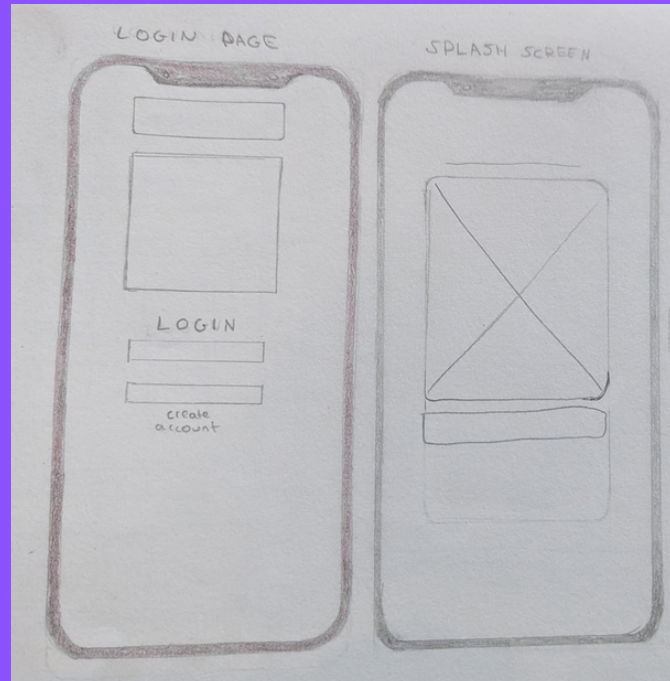
**Spotify**

- Colour pallet
- Playlists and collections
- Layout of features

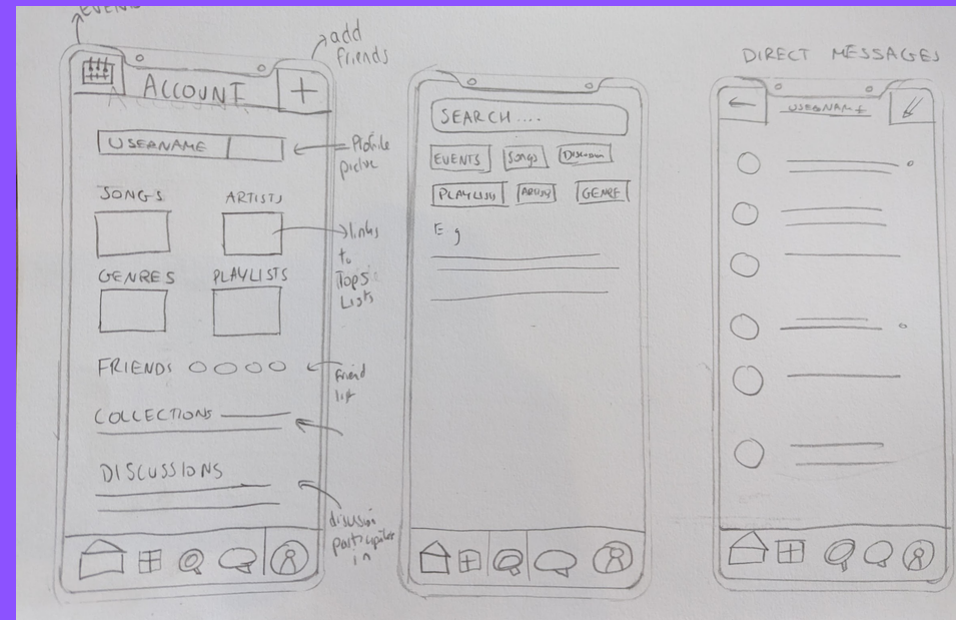


# Initial Designs

Now that I had decided what features I wanted to include in my app I started creating basic sketches of my app layout.

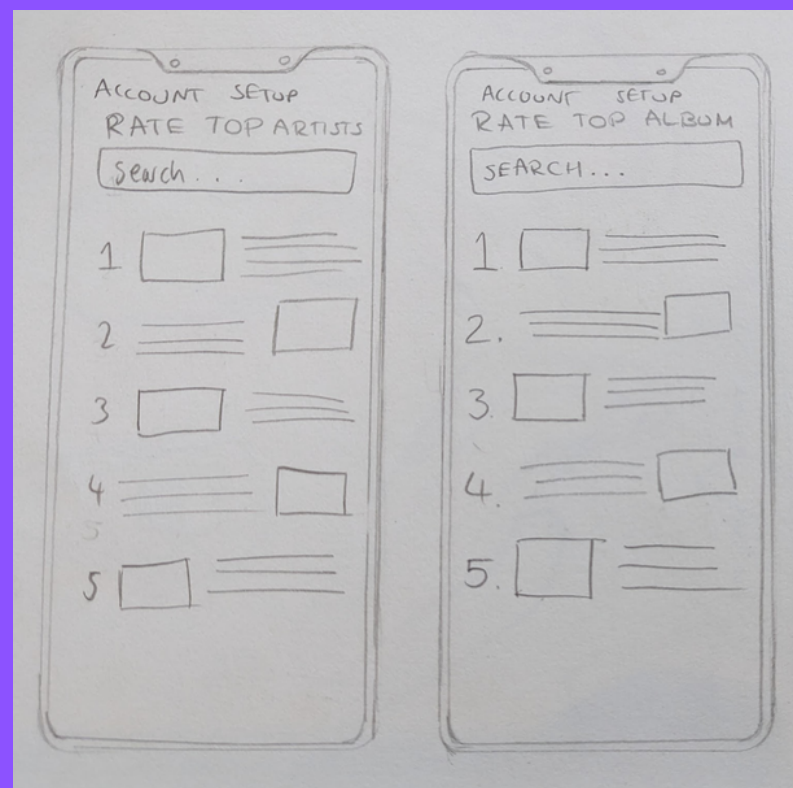


The login layout of the app was intentionally designed to be simple and user-friendly, aligning with the conventions commonly observed in many apps. Additionally, I incorporated a splash screen feature, as it has become a standard feature in modern app design. This introductory screen provides an engaging and visually appealing experience for users when they first access the app.

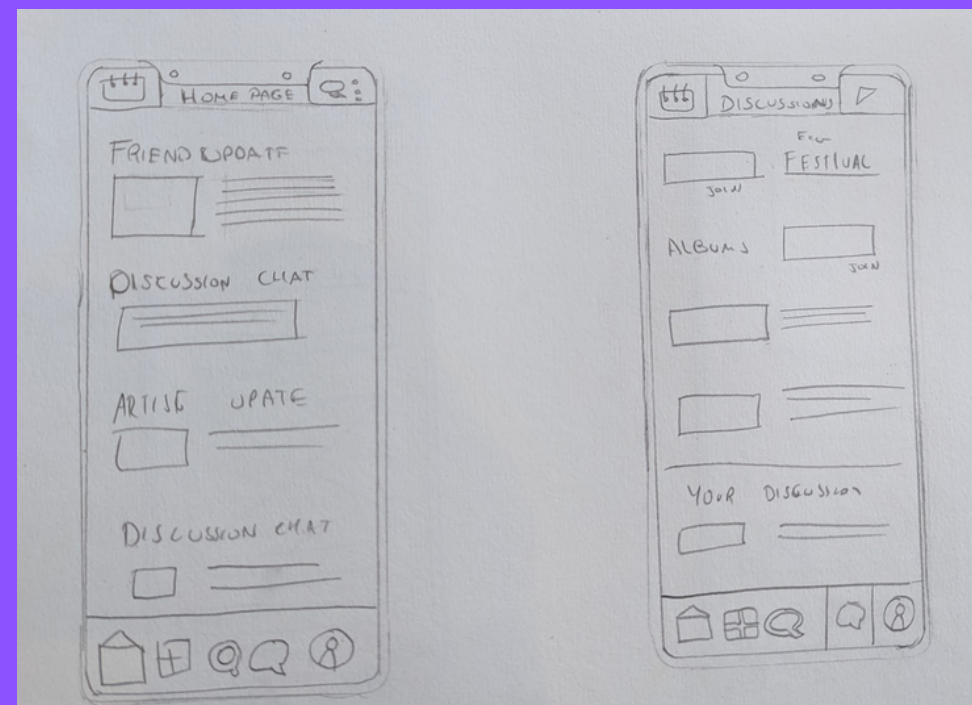


In terms of navigation, I implemented a tab bar at the bottom of the interface to ensure easy access to key sections of the app. This tab bar houses the home page, adding collections, a search bar, a discussions page, and an account page. By organising these features in the tab bar, users can effortlessly navigate between different sections and access desired functionalities.

The top navigation section is dedicated to adding users and events, providing a straightforward way for users to initiate these essential interactions.



Drawing inspiration from Spotify's popular "Top Songs" feature at the end of each year, I integrated a similar concept into the account setup process. During this setup phase, users are prompted to input their top 5 albums, songs, artists, and genres. This allows users to personalise their profiles and provides valuable insights into their music preferences, enhancing the overall user experience.



The home page serves as a central hub, providing a snapshot of various aspects of the app. It offers quick updates and highlights from different sections, allowing users to stay informed about the latest activities and events within the app. This centralised view ensures that users have a comprehensive overview of what's currently happening and can quickly explore into the areas that interest them most.

In the discussions page, users are presented with topic categories to facilitate easy identification and participation in group discussions. Additionally, there is a section dedicated to ongoing chats, allowing users to quickly access and engage in conversations they are already involved in. This intuitive design promotes community engagement and fosters a sense of belonging.

# Wireframes

After creating initial designs, I used Figma to start laying out where everything should go on my app. As my app has many different features, I needed many pages so I sought out figuring out the functionality.

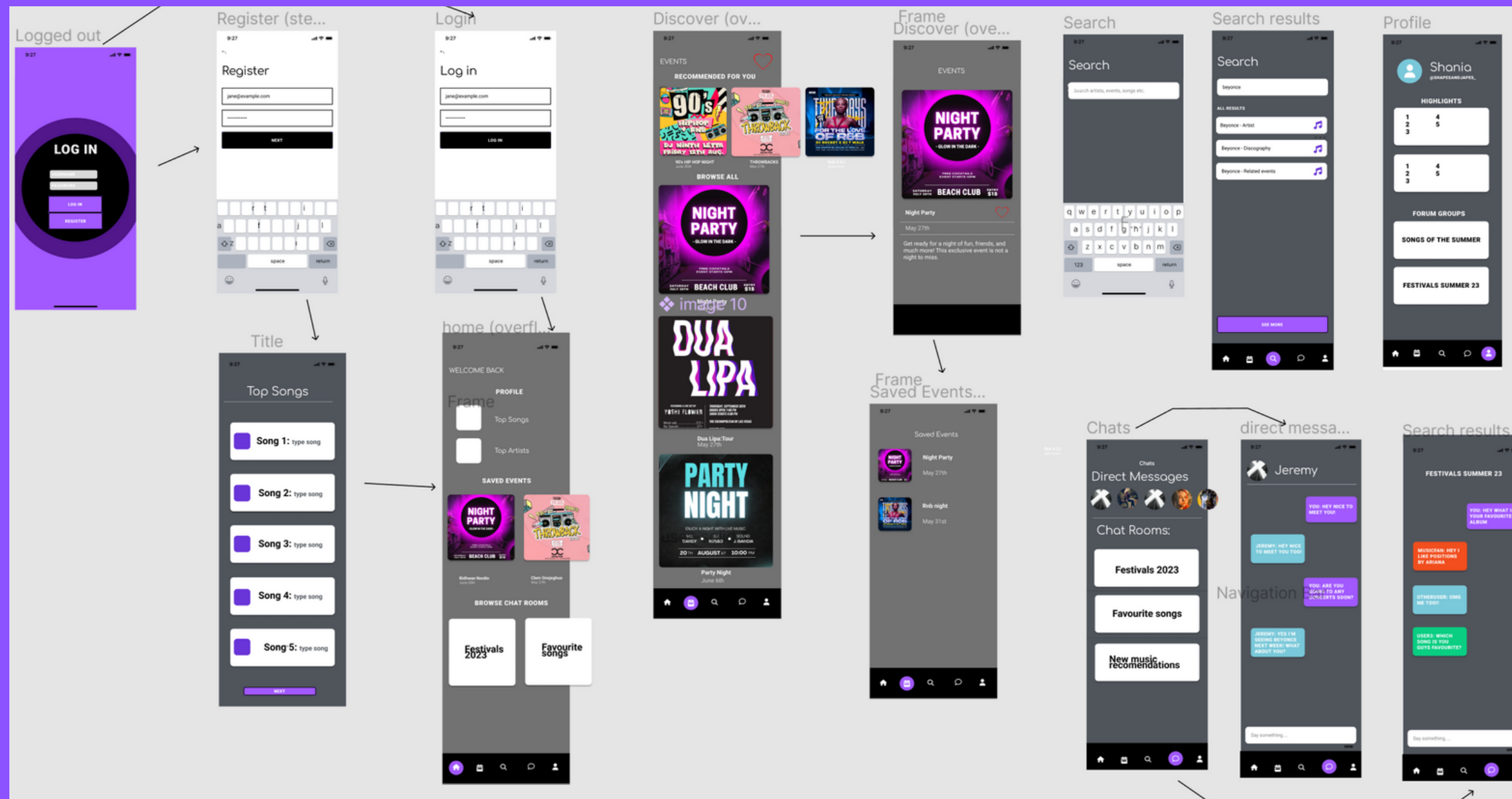
The initial part of my app begins with a user-friendly login page that offers two options: existing users can log in, while new users can easily register. Upon successful login, users are seamlessly directed to their personalised home page, where they can explore a range of exciting features tailored to their preferences. These features include the user's Top 5s, saved events, and active chat rooms.

For users who choose to register, the app guides them through a streamlined account setup process. As part of this setup, users are prompted to create their personalized Top 5s, reflecting their favourite artists or songs. Once the account setup is complete, users are swiftly taken to their home page.

The app's intuitive navigation system ensures easy access to various sections, with tabs conveniently arranged from left to right. The "Events" tab allows users to effortlessly search for and save upcoming events that are tailored to their interests.

The "Search" tab empowers users to explore and discover new singers, expanding their musical repertoire. The "Chat Rooms" tab serves as a vibrant hub for direct user-to-user communication and engaging group discussions, fostering connections and sharing musical experiences.

Finally, the "Profile" tab offers users a comprehensive overview of their personal data, providing insights into their music preferences, saved events, and chat activity, empowering them to track their musical journey and interactions within the community.



I made the strategic decision to streamline the app's user interface by removing certain features from the top navigation bar. The intention behind this modification was to enhance the user experience by reducing complexity and making the app more intuitive to navigate. Recognising the significance of the "Events" page as a primary feature, I repositioned it to the bottom tab bar for prominent accessibility and seamless user interaction.

In terms of collections, I re-evaluated the app's focus and determined that prioritising the social aspect through visually appealing collections would detract from the core objective of facilitating music discovery. Consequently, I opted not to incorporate a feature for creating collections. Instead, users are provided with the opportunity to curate and update their "Top 5" collection, allowing for the personalised expression of their favourite artists or songs. This streamlined approach strikes a balance between user customisation and overall app simplicity.

# Key Features

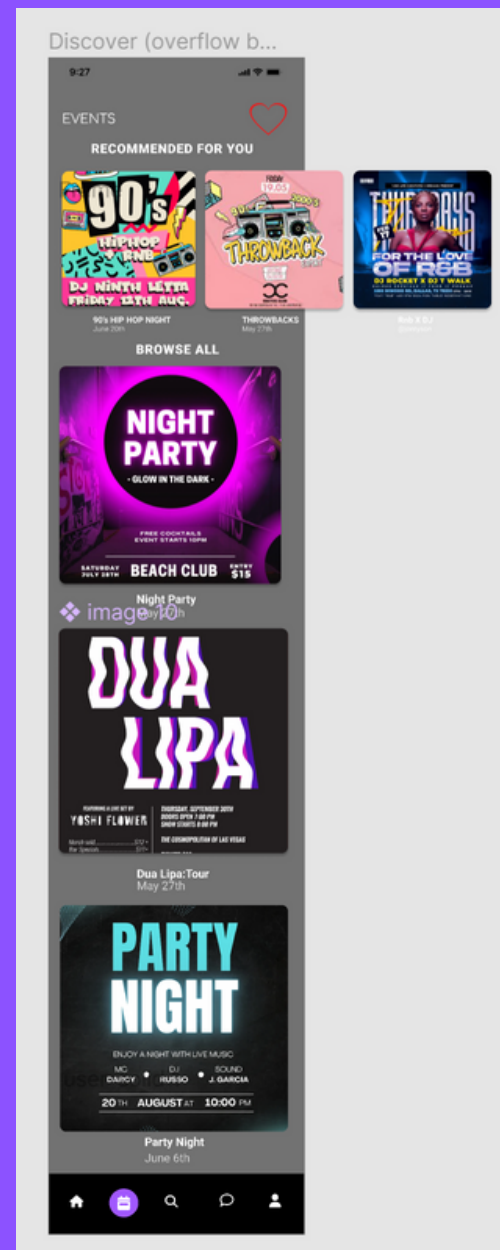
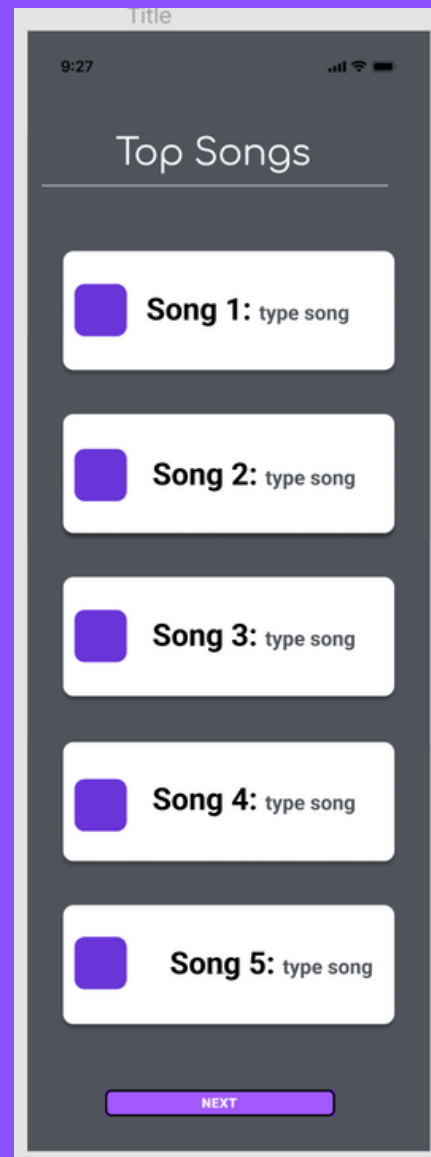
To allow this app to stand out and engage users, I wanted to integrate features from other apps. Here are the key features.

The 'Chat Room' page is also a key page in my app as it allows for users to be social and gain music and event recommendations. There are group chats as well as private discussions on this app depending on users' preferences. The group discussions are curated by topic so users can decide which chats they would like to participate in or if they want to make their own themes. The private chat is curated for general chat and making friends.

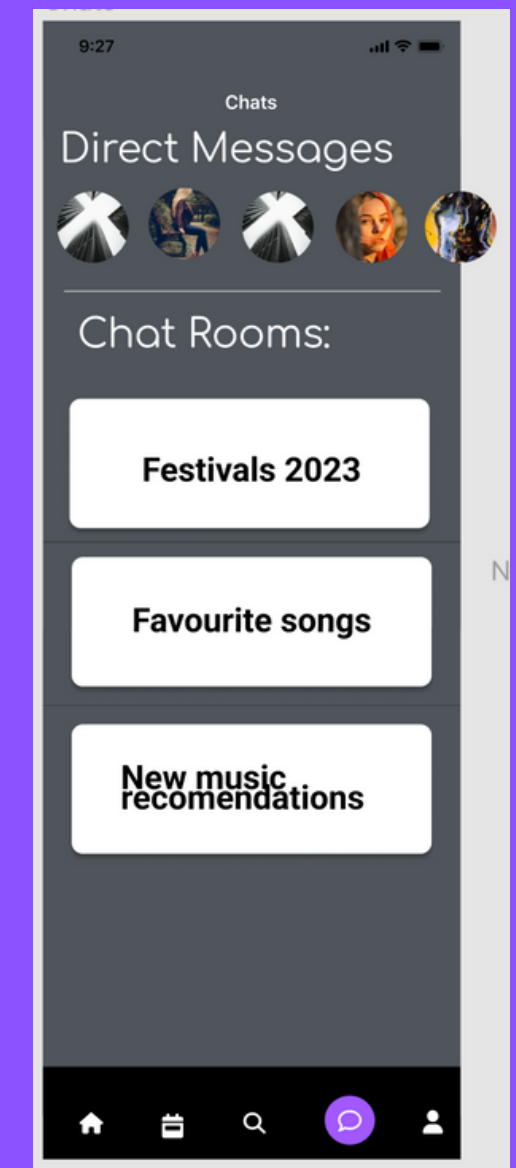
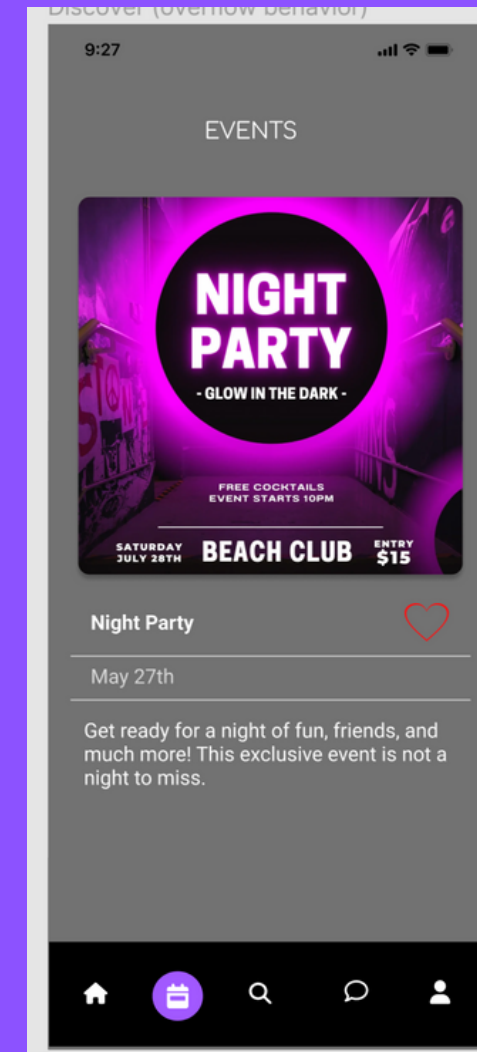
The 'Events' page is one of the important pages on the app as it allows users to see what events are being held in their local area. Users can explore all events but also have a recommended section, which allows for a quick search to see events tailored to them. Users could discuss and share events with friends and in bigger discussions.



When users initially registered they are asked to fill out their top songs, artists, albums and genres in order to set up a profile. This allows other users on the app see what type of music a user is interested in which enables discussions and socialising. Users can update their profile at any time.



When a user clicks on an event they can have more event details, if a user would like to attend they can like the event which automatically saved the event to the saved events tab at the top of the Events page. Users can track what events they like for future reference and the app can learn users preferences.



# Code

Here are some code snippets of work I found particularly difficult

In order to establish a group chat conversation, I aimed to develop several chatbots with distinct outputs and response timings. I utilised the dispatched features and set the outputs to occur randomly and at intervals of a few seconds. Users had the freedom to respond whenever they wished. To ensure clarity and simulate a typical text message chain, I implemented a mechanism that automatically scrolled the phone screen to display the most recent message whenever a new message from the chatbot was sent.

```
7
8
9 import SwiftUI
10
11 struct ChatRoomView: View {
12     var roomName: String
13     @State private var messages: [Message] = []
14     @State private var newMessage: String = ""
15     @State private var scrollToBottom = false // Track whether to scroll to the bottom
16
17     var chatbots: [Chatbot] = [
18         Chatbot(name: "Musicfan", color: Color.blue),
19         Chatbot(name: "Festivallover", color: Color.green),
20         Chatbot(name: "musicmark", color: Color.orange)
21     ]
22
23     var body: some View {
24         ZStack {
25             Color(red: 0.2, green: 0.2, blue: 0.2).edgesIgnoringSafeArea(.all)
26             VStack {
27                 ScrollView {
28                     ScrollViewReader { scrollViewProxy in
29                         LazyVStack {
30                             ForEach(messages) { message in
31                                 SpeechBubble(message: message)
32                                     .id(message.id)
33                             }
34                         }
35                     }
36                     .onChange(of: messages, perform: { _ in
37                         if scrollToBottom {
38                             scrollViewProxy.scrollTo(messages.last!.id, anchor: .bottom)
```

```
77     }
78
79     let userMessage = Message(sender: "You", content: newMessage, color: Color.purple)
80     messages.append(userMessage)
81
82     newMessage = ""
83 }
84
85 private func startChatbotTimers() {
86     for chatbot in chatbots {
87         DispatchQueue.main.asyncAfter(deadline: .now() + Double.random(in: 5.0...7.0)) {
88             let chatbotMessage = Message(sender: chatbot.name, content:
89                 generateRandomMessage(), color: chatbot.color)
90             messages.append(chatbotMessage)
91             checkChatbotMessages()
92         }
93     }
94
95     private func checkChatbotMessages() {
96         let totalMessages = chatbots.count
97         if messages.filter({ $0.sender != "You" }).count >= totalMessages {
98             DispatchQueue.main.asyncAfter(deadline: .now() + Double.random(in: 5.0...7.0)) {
99                 let chatbotMessage = Message(sender: chatbots.randomElement()!.name,
100                     content: generateRandomMessage(), color:
101                     chatbots.randomElement()!.color)
102                 messages.append(chatbotMessage)
103                 checkChatbotMessages()
104             }
105         }
106     }
107 }
```

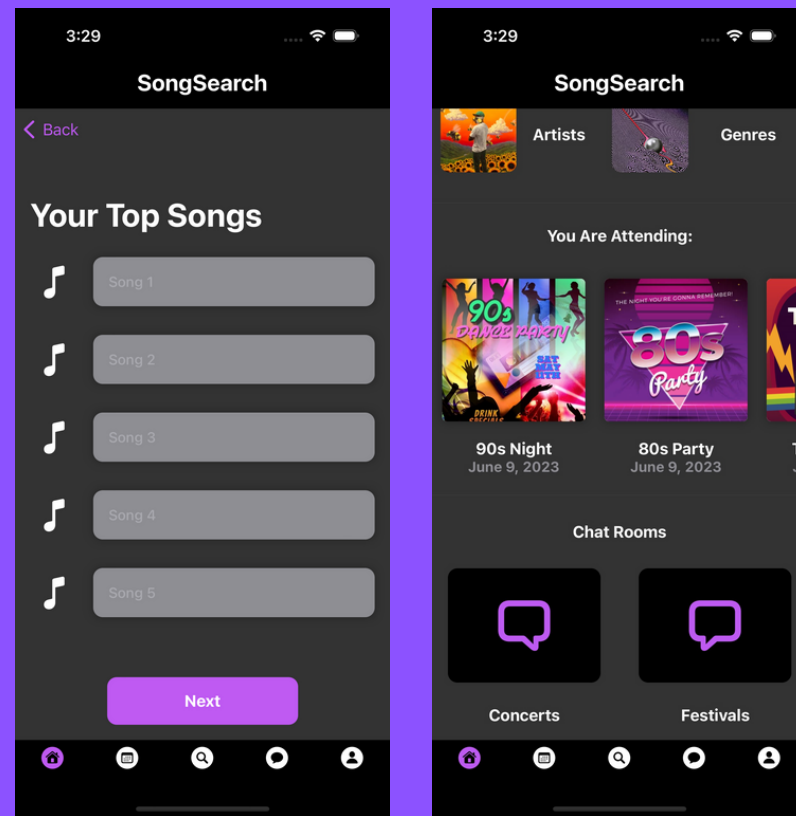
```
7
8 import SwiftUI
9
10 struct SavedEventsView: View {
11     @EnvironmentObject var savedEventsViewModel: SavedEventsViewModel // Access the
12     SavedEventsViewModel as an environment object
13
14     var body: some View {
15         ZStack {
16             Color(red: 0.2, green: 0.2, blue: 0.2)
17                 .ignoresSafeArea()
18
19             NavigationView {
20                 List(savedEventsViewModel.savedEvents, id: \.title) { event in // Use the
21                     savedEventsViewModel's savedEvents array to populate the list
22                     NavigationLink(destination: EventDetailsView(event: event, description:
23                         nil)) { // Navigate to the event details view
24                         HStack {
25                             if let posterImage = event.poster {
26                                 Image(uiImage: posterImage)
27                                     .resizable()
28                                     .aspectRatio(contentMode: .fit)
29                                     .frame(width: 60, height: 60)
30                             }
31
32                             Text(event.title)
33                                 .foregroundColor(.white)
34                         }
35                     }
36                 }
37             }
38         }
39     }
40 }
```

To enable the search bar to have multiple different categories, I implemented separate arrays for each category and integrated them into the ExploreView. Each category is associated with a distinct button, and when a button is clicked, the corresponding list of elements is displayed. Additionally, the search feature filters out elements based on specific search criteria.

```
7
8
9 }
10
11 var filteredSingers: [String] {
12     let lcRnbSingers = listRnbSingers.map { $0.lowercased() }
13
14     return searchText.isEmpty ? lcRnbSingers : lcRnbSingers.filter {
15         $0.contains(searchText.lowercased())
16     }
17 }
18
19 var filteredEvents: [String] {
20     let lcEvents = listMusicEvents.map { $0.lowercased() }
21
22     return searchText.isEmpty ? lcEvents : lcEvents.filter {
23         $0.contains(searchText.lowercased())
24     }
25 }
26
27 var filteredGenres: [String] {
28     let lcGenres = listMusicGenres.map { $0.lowercased() }
29
30     return searchText.isEmpty ? lcGenres : lcGenres.filter {
31         $0.contains(searchText.lowercased())
32     }
33 }
34
35 var filteredSongs: [String] {
36     let lcSongs = listRnbSongs.map { $0.lowercased() }
37
38     return searchText.isEmpty ? lcSongs : lcSongs.filter {
39         $0.contains(searchText.lowercased())
40     }
41 }
```

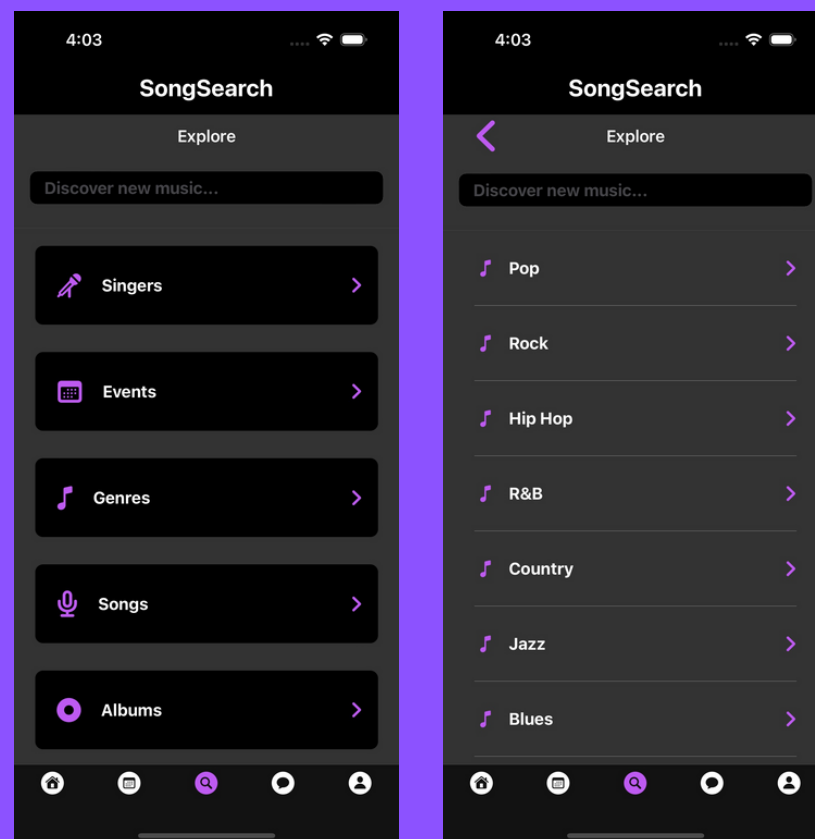
To save events for users to see in a different view I created an instance of SavedEventsViewModel, which is accessed as an environment object using @EnvironmentObject. This means that the view expects to receive an instance of SavedEventsViewModel when it is used. The NavigationView contains a List that displays the saved events. The data for the list is obtained from the savedEventsViewModel.savedEvents array, and each event is rendered as a row in the list. For each event in the list, a NavigationLink is created. When a user taps on a row, the NavigationLink navigates to the EventDetailsView, passing the selected event as a parameter. This allows users to view the details of the selected event.

# Final Designs

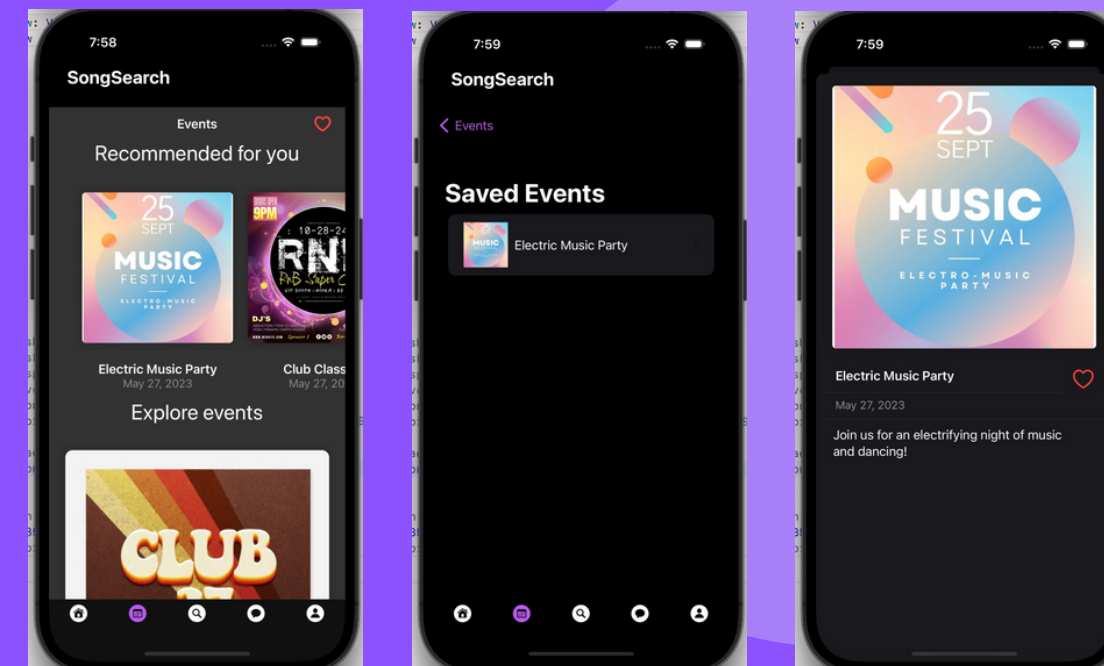


Here is what the Home Page Looks like. It includes the option to change the top 5 Lists, the users Saved Events, and live chat rooms. The list feature is also shown.

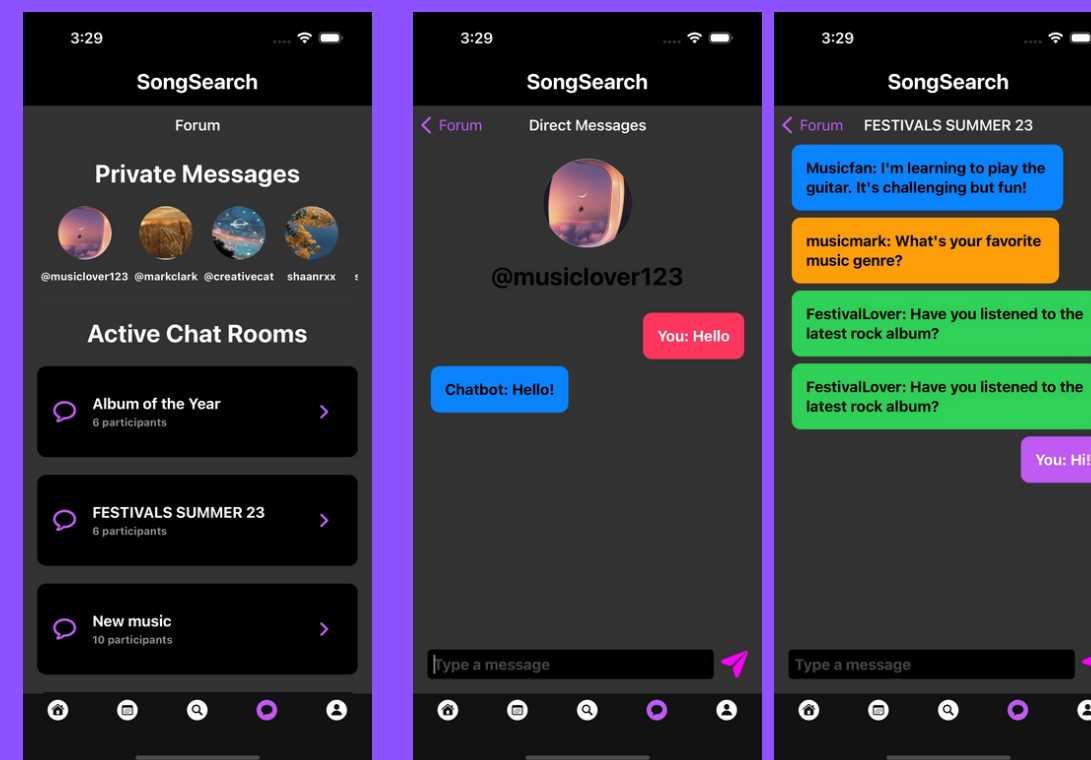
The Search feature has 5 different categories and 5 different lists of data that a user can search for.



Events

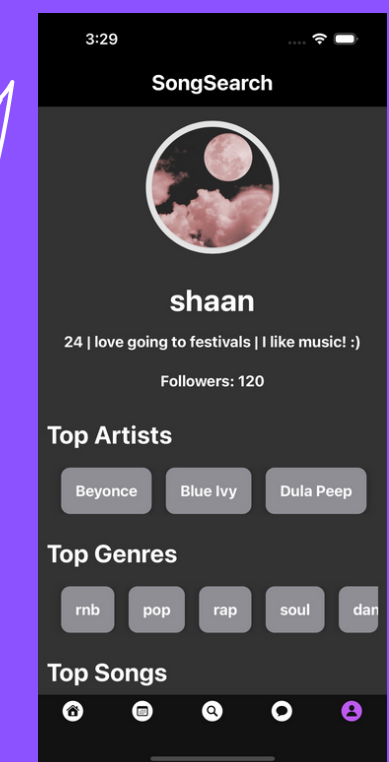


I kept the design of the Events page identical to the original design and used sheets to show event details



Here are the the private and group chat rooms and how the two options are laid out in the forum page. The number of participants are also shown.

The account page is slightly different as I created a horizontal scroll for each top 5 list instead



# Limitations/ Challenges

- **User Testing:** If I had additional time, I would have conducted user testing with the intended demographic to evaluate the app's performance, identify strengths, and hear areas for improvement. User testing is crucial because it provides valuable insights into how users interact with the app, uncovers usability issues, and helps refine the user experience.
- **Permission:** In order to display, Songs, Artists, and Events, the app would need permission to add these into the app. An alternative would be working with another app such as Spotify to link together limit problems.
- **Number of users:** To create an app that enables users to meet new people, there needs to be a certain amount of users signed up. To ensure maximum efficiency of these features there would need to be lots of promotion for this app.
- **Community Moderation:** Maintaining a healthy and respectful community requires active moderation and addressing any inappropriate or harmful content. Implementing robust community guidelines and effective moderation tools can help solve potential challenges.
- **User Empowerment:** Encouraging users to actively participate in maintaining a positive community environment. Implementing features that allow users to block or mute others, the customization of content visibility settings and the ability to provide feedback on reported content. Empowering users helps to create a sense of ownership and fosters a self-regulating community.

# Improvements

- **Overall,** I am happy with the way my completed app looks and the functionality. I am happy with my chosen colour pallet and I managed to add all the features I originally set out to do. However, if I had more time to improve the app I would do the following:
  - **Add a create room feature,** where users can set up their own chat rooms. Additionally I would add an admin feature so users can be in control of who stays in a room which can help limit inappropriate content.
  - **Add a review section,** where users can rate certain events, albums, and songs as well as having a top 5. This would help the algorithm know what events to recommend to the users and help other users gain more knowledge and opinions.
  - **Add a collections feature.** I would like for users to still have a creative section in the app like a visual playlist. I could potentially work on the top 5 feature to integrate the two together to make a more interactive feature.

# References

- Besseny, A. (2020). Lost in spotify: folksonomy and wayfinding functions in Spotify's
- Cha, M.H. and Woo-Kyung, L., 2021. A Study on Interface Design of Meditation App for Improving Mental Health of Adolescents. *Journal of Digital Art Engineering and Multimedia*, 8(2), p.153.
- Einfeld, Henriette, and Felix Kristallovich. "The rise of dark mode: A qualitative study of an emerging user interface design trend." (2020).
- Eriksson, M. and Johansson, A., 2017. Tracking gendered streams. *Culture unbound. Journal of Current Cultural Research*, 9(2), pp.163-183.
- Freeman, S., Gibbs, M. and Nansen, B., 2022. 'Don't mess with my algorithm': Exploring the relationship between listeners and automated curation and recommendation on music streaming services. *First Monday*.
- Hakobyan, L. and Saha, R., 2021. The impact of dark mode on the visual attractiveness of social media postings: A users' perception study based on Facebook.
- Jackson, M.S., 2023. *Spotify: The Addiction*.
- Lehtiniemi, A. and Holm, J., 2013. Designing for music discovery: Evaluation and comparison of five music player prototypes. *Journal of New Music Research*, 42(3), pp.283-302.
- Pedersen, L., Einarsson, S., Rikheim, F., & Sandnes, F. (2020). User Interfaces in Dark Mode During Daytime – Improved Productivity or Just Cool-Looking?. *Lecture Notes In Computer Science*, pp. 178-187.
- Snickars, P., 2017. More of the same–on Spotify radio. *Culture Unbound. Journal of Current Cultural Research*, 9(2), pp.184-211.
- Werner, A., 2020. Organizing music, organizing gender: algorithmic culture and Spotify recommendations. *Popular Communication*, 18(1), pp.78-90. [Original source: <https://studycrumb.com/alphabetizer>]